Visualizing Discovery Paths

Simon Alm Liljevall and Karl Jansson

DEPARTMENT OF DESIGN SCIENCES
LUND UNIVERSITY

2016

MASTER THESIS

Visualizing Discovery Paths

Design & Implementation

Simon Alm Liljevall and Karl Jansson

LUND
UNIVERSITY

Visualizing Discovery Paths

Design and implementation

Published by
Department of Design Sciences
Faculty of Engineering LTH, Lund University
P.O. Box 118, SE-221 00 Lund, Sweden

Subject: Interaction Design (MAMM01)
Division: Division of Design Sciences
Supervisor: Joakim Eriksson
Co-supervisor: Tobias Åström (Qlik)
Examiner: Jonas Borell

Abstract

Business intelligence (BI) is all about making decisions. The purpose of many business intelligence tools is to improve these decisions by making data easy to visualize and understand. Qlik Sense is a BI tool developed by the company Qlik which allows users to visualize and explore data by filtering it (called making selections). At the time of this project Qlik Sense did not support visualization of a user's selection history. Hence, users were not able to share their path to discovery. This project aimed to find and implement the most suitable way of solving that problem. Here we show, taking user and expert feedback into consideration, that a suitable way of visualizing discovery paths is done as a tree representation of the selections made by the user. In order to create the final product an iterative approach based on user-centered design (UCD) was used. Initially a brainstorming session was held which resulted several concepts to help produce low-fidelity prototypes. Based on heuristic evaluations and user tests, these prototypes resulted in the final product implemented as a part of Qlik Sense. Furthermore, the result of this project forms a well working solution and a solid foundation for further development towards a complete product.

Keywords: business intelligence, Qlik Sense, user-centered design, discovery paths

Acknowledgments

We would like to thank all of the people who have been part of this project.

Additionally, we would like to thank Joakim Eriksson for helping throughout the entirety of this project. Your suggestions and feedback has been very valuable to us during both development of the product as well as during the documentation of this project.

Furthermore, we would like to thank Tobias Åström, Imir Useini and Quan Ho and the rest of the involved staff members at Qlik for giving us this opportunity and for giving us guidelines and answering our questions.

Lastly, I (Simon) would like to give a big shoutout and an even bigger thank you to my fiancé, Mathilda Rosén (soon to be Mathilda Rosén Liljevall), for sticking with me through thick and thin. You help me get through the toughest of times with your perfect personality blend of warmth and sharpness. You truly are the love of my life!

Lund, May 2016

Simon Alm Liljevall & Karl Jansson

Table of contents

List of acronyms and abbreviations

API	application programming interface
BI	business intelligence
CSS	Cascading Style Sheet
D3	Data Driven Documents
Hi-Fi	high fidelity
HTML	Hyper Text Markup Language
JS	Javascript
Lo-Fi	low fidelity
SDK	software development kit
UCD	user centered design

1 Introduction

To get a better understanding of Qlik Sense and the goal of the thesis we recommend you to read section 2.1 about Qlik Sense before continuing to next section.

1.1 Background

Qlik was founded in 1993 by Björn Berg and Staffan Gestrelius with the vision to create an entirely new piece of software that truly mimicked the way that the brain works in order to ensure the most intuitive user experience possible (Qlik.com).

Qlik has since then grown to a multinational company, providing their agile business intelligence solutions to approximately 35000 customers worldwide. Business intelligence (BI) is all about making decisions. The purpose of many business intelligence tools is to improve these decisions by making data easy to explore. In Qlik Sense this is done by letting users filter (called making selection) and create visualization of data in an intuitive way. This helps users to understand and make the discoveries of the data. However, at the time of this project there was no way to visualize in which order these selections were made during a discovery and thus the user could not share their path to insight.

1.2 Purpose

The purpose of this project was to investigate the possibilities of how to visualize and share a user’s path to insight in Qlik Sense by using multiple interaction design methods.

1.3 Problem Definition

In order to produce the most suitable solution to the problem the following questions were used as guidelines during the development process.

1.3.1.1 Primary question

How is a user's path to discovery best visualized in Qlik Sense?

1.3.1.2 Complementary questions

What information should a discovery path contain?

Which limitations currently exist in Qlik Sense for this visualization to be fully implemented?

Can this type of functionality be implemented as an extension?

Which are the advantages and disadvantages of this way of visualizing?

Which type of visualization is considered intuitive by users?

1.4 Project Scope

Since development for Qlik Sense requires a large amount of knowledge about the product and how it works on the inside, the goal of this thesis is to produce a concept and product based on interaction design theory. The final product is implemented as an extension in Qlik Sense, showing the main functionality of the concept.

1.5 Limitations

Due to Qlik policy their customers could not be part of the development process of the conceptual extension. In order to make the design process user centered, non-Qlik-customers were considered as potential users throughout the project. With the motivation - if users with no previous knowledge of Qlik Sense find the extension intuitive, then so will users with previous knowledge.

2 Technical Background

This section gives a brief description of the most crucial technical tools which were used during this thesis.

2.1 Qlik Sense

Qlik Sense is a large and complex software with various fields of application. This section explains the most relevant parts of Qlik Sense.

Briefly, Qlik Sense is an interactive BI tool to visualize data. By loading data into Qlik Sense a user can create an application that contain sheets to visualize the data in various ways. The user can interact with the sheets by making selections which may result in new insights and discoveries.

2.1.1 Apps

When starting Qlik Sense Desktop the menu in Figure 1 are displayed.

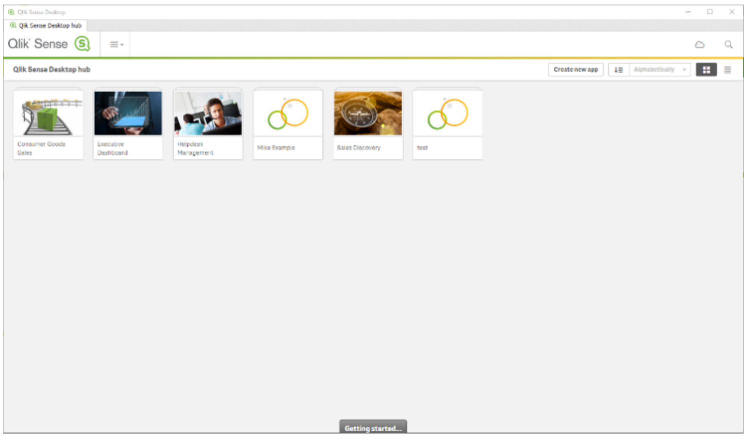


Figure 1: Qlik Sense start menu

The menu shows a user’s saved applications and also allows a user to create new ones. When doing so, the user loads the data which he/she want to visualize and explore. The data can be anything from sports to company sales statistics. The user can then create sheets to visualize the data in the application. Concisely one could say that an application is a map containing sheets to visualize the application-data.

2.1.2 Data

As mentioned above each application contains data. This data contains multiple fields, a field has a title and contains values. A field could for example have the title Countries and hold values such as Sweden, Norway, France etc. Qlik Sense supports more complex data structures but this is not important for the understanding of this thesis and is therefore not taken in consideration here.

2.1.3 Sheets

Sheets in Qlik Sense offers various ways of visualizing data such as bar charts, line charts, pivot tables and maps. A sheet can hold one or many different charts. To create a sheet the user simply drags a chart from the left menu into the edit area and choose what app-data the chart should visualize.

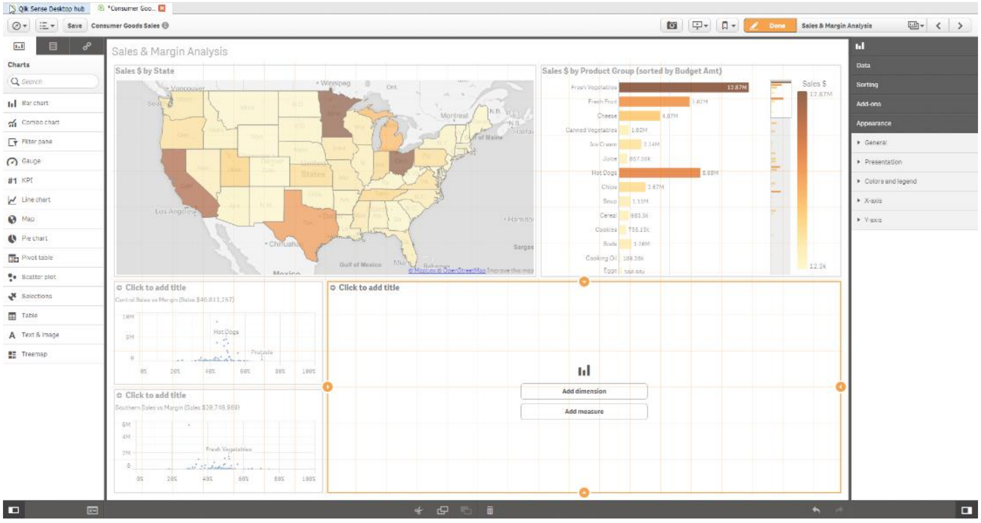


Figure 2: Qlik Sense sheet editor

The Qlik Sense editor also has other functionalities such as sort algorithms, title editing and much more.

2.1.4 Selection

When a sheet is created the user can start to analyze the data in the application. One of the main functionalities in Qlik Sense is selection. Selections allow users to filter data. Each selection contains a fieldname and the field-values the user want to analyze further. A selection is made by interacting with the charts or by searching. The sheets in the application are dynamically changed when a selection or de-selection is made, allowing the user to freely explore smaller sets of data. In Figure 3 a sheet with a table over sales representatives and a bar chart visualizing their total sales amount is shown.

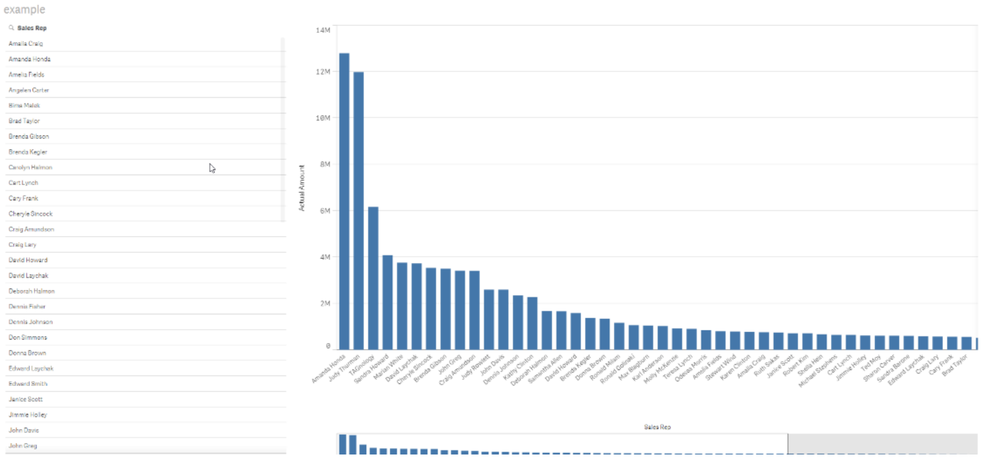


Figure 3: Bar chart over sales representatives and their sales amount

If a user makes selections of sales representative in the left table the bar chart is automatically changed. This is called linked views (Wills, 2008). In Figure 4 four sales representatives have been selected and as expected, only the selected sales representatives are shown in the bar chart.

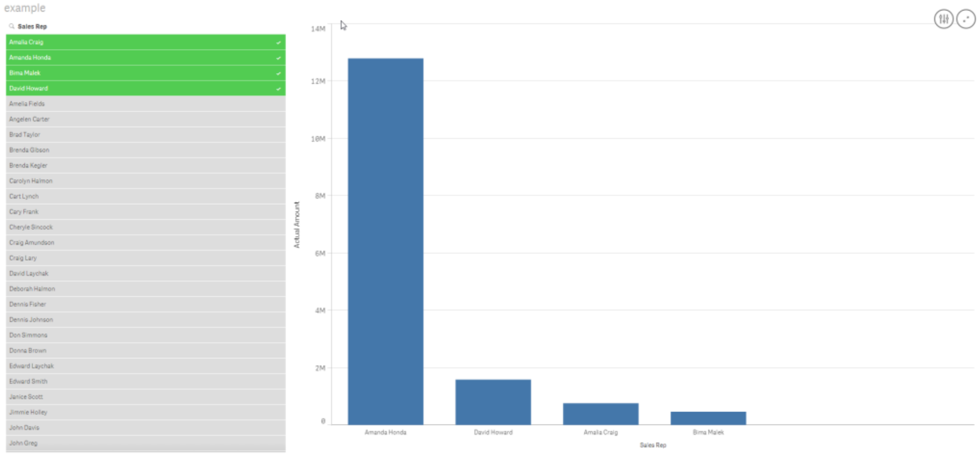


Figure 4: Updated bar chart when four sales representatives were selected.

2.1.5 Extensions

Qlik Sense allows users to create their own visualizations of data with extensions. An extension is mainly written in Hyper Text Markup Language (HTML), Cascading Style Sheets (CSS) and, Javascript (JS) with Qlik’s own application programming interface (API) and software development kit (SDK). The extension can also be integrated with other systems and visual libraries such as Data Driven Documents (D3).

3 Theoretical Background

This section gives a brief description of the most crucial theoretical frameworks which were used for this thesis. The goal of this section is to give the reader basic knowledge regarding theoretical fundamentals before describing the process.

3.1 User-centered design (UCD)

UCD is a broad term to describe design processes in which end-users influence how a design takes shape. It is both a broad philosophy and variety of methods. There is a spectrum of ways in which users are involved in UCD but the important concept is that users are involved one way or another. Studies have shown that UCD leads to developing more usable satisfying design compared to other approaches (Abras, Maloney and Preece, 2004).

Furthermore, Gould and Lewis suggest three principles of user-centered design (Gould and Lewis, 1985, pp. 300):

- *Early focus on users and tasks*. First, designers must understand who the users will be. This understanding is arrived at in part by directly studying their cognitive, behavioral, anthropometric, and attitudinal characteristics, and in part by studying the nature of the work expected to be accomplished.
- *Empirical measurement*. Second, early in the development process, intended users should actually use simulations and prototypes to carry out real work, and their performance and reactions should be observed, recorded and analyzed.
- *Iterative design*. Third, when problems are found in user testing, as they will be, they must be fixed. This means design must be iterative. There must be a cycle of design, tests and measure, and redesign, repeated as often as necessary.

3.2 Participatory design

Participatory design is a development strategy that involves potential users in the design process. In some cases these users are asked to review a product in development and provide feedback; in other cases they are actively involved in generating design concepts (Barnum 2011, p. 58).

3.3 Heuristic evaluation/Expert review

A heuristic evaluation refers to an expert review, or inspection, of a product, using a specific set of guidelines or, in some cases, in a more loosely defined way (Barnum 2011, p. 58). This can be done by usability experts or by double expertise people; in both usability and the product domain. Heuristics are a set of principles used by the experts to inspect an interface in search of violations of the heuristics (Barnum 2011, p. 61).

3.4 Brainstorming

Brainstorming is an activity used to generate ideas in groups and was originally introduced by Alex Osborn in 1957 (Henningsen and Henningsen, 2013). The activity aims to produce as many ideas as possible in the given amount of time and is claimed to do exactly that in a more efficient way than normal group discussions. Brainstorming emphasizes the importance of positivity and being open minded to every suggestion brought up during the session. It also focuses on combining ideas and expanding on ideas from others.

The topic of brainstorming and its efficiency compared to nominal groups is an old and largely discussed topic where research has shown different results throughout the years. Research made by the Henningsen brothers summarizes the topic in the following way:

Even if brainstorming groups are not more productive than nominal groups, it remains possible that the difference in productivity is not so much a difference in the quantity of ideas produced but rather an artifact of the amount of time required to produce them. If so, the concomitant socioemotional gains in brainstorming groups may in the long run offset the slightly greater increments in time on task that brainstorming may require. It is also possible that brainstorming groups may, in fact, be more productive than nominal groups. If nominal groups are less likely to persist at idea-generation tasks than brainstorming groups, over time, the brainstorming groups may demonstrate superiority in terms of the number of ideas produced (Henningsen and Henningsen, 2013, p. 53).

3.5 Prototyping

A prototype is a manifestation of design that allows stakeholders to interact with it and to explore its suitability; it is limited in that a prototype will usually emphasize one set of product characteristics and de-emphasize others (Rogers, Sharp, and Preece, 2011). The activity of building prototypes encourages reflection in design (Schön 1984).

Prototypes can come in different forms and level of completeness. In this thesis two different prototypes were used as part of the design process; low-fidelity (lo-fi) prototypes and a high-fidelity (hi-fi) prototype.

3.5.1 Low-Fidelity prototypes

Lo-fi prototypes are prototypes that do not look very much like the final product. For example, lo-fi prototypes use materials that are very different from the intended final version, such as paper and cardboard rather than electronic screens and metal. Low fidelity prototypes are useful because they tend to be simple, cheap, and quick to produce. This also means that they are simple, cheap, and quick to modify so they support the exploration of alternative designs and ideas (Rogers, Sharp, and Preece, 2011, pp. 391-392).

3.5.2 High-Fidelity prototypes

High fidelity prototyping uses materials that you would expect to be in the final product and produces a prototype that looks much more like the final thing. For example, a prototype of a software system developed in Visual Basic is higher fidelity than a paper based mock-up. High fidelity prototyping is useful for selling ideas to people and for testing out technical issues (Rogers, Sharp, and Preece, 2011, pp. 395).

3.6 Personas

Personas and their use are described differently depending on who you ask. In this thesis the definition of John Pruitt is used (Pruitt and Adlin, 2005, p. 3).

Personas are detailed descriptions of imaginary people constructed out of well-understood, highly specified data about real people.

Furthermore, it has been suggested (Pruitt and Adlin, 2005, pp. 4) that when you use data to create personas, and use personas in a thoughtful way during the product development process, you will:

- Increase your products' usability, utility, and general appeal.
- Streamline your teams' processes and improve your colleagues' abilities to work together.
- Enable your company to make business decisions that help both your company and your customers.
- Improve your company's bottom line.

3.7 Scenario based testing

Scenarios are informal narrative descriptions used to test a design. The scenario should help the tester work through a complex problem or test system (Carroll, 2000). A good scenario has five characteristics (Kaner, 2003):

- Based on a story how the program is used.
- The story is motivating.
- The story is credible.
- The story involves complex use of the program, or complex environment or a complex set of data.
- The test results are easy to evaluate.

4 Initial phase

This was the initial phase of the project and also where we spent much time preparing the project, learning the context (Qlik Sense) and investigated how customers use Qlik Sense.

4.1 The typical Qlik Sense user

In order to understand what to develop we needed to understand how customers used Qlik Sense. The easiest way of doing this would be to talk to the customers directly. However, as mentioned above, due to Qlik policy we did not have access to the costumers. Thus, we had to take a different approach and luckily Qlik had developed personas.

A meeting with a team manager regarding these personas helped in eliciting which ones to take into consideration when developing the product. This resulted in two personas; one who is a data analyst and one who is a presenter.

4.2 What is a discovery path and what information does it contain?

To be able to even have a brainstorm session we first had to get a better understanding of what a discovery path is. After testing Qlik Sense we quickly understood what the main goal of the project was; to visualize the selection history.

As mentioned earlier, in order to explore data a user makes selections. These selections contains the field and the values that the user wants to analyze further. A discovery path is, in other words, a sequence of previously selected fields and values.

4.3 Making an extension or not?

This was one of the earliest questions which had to be answered. Is an extension a suitable approach to this problem? After a brief discussion with the project supervisor who had several years of experience with Qlik Sense, it was concluded that taking the extension approach was the most suitable way. The reasoning being that Sense is a complicated program and thus it takes a long time learning it. The extension framework is built specifically to let people develop their own customization for the program.

Making an extension also means that you are not able to change the core of Qlik Sense. Whether this would be limiting during development was uncertain at this point of the project. However, taking all of these things into consideration, an extension seemed to be the most suitable approach for this project.

5 Brainstorming

This section describes the idea generation. This was done through brainstorming and resulted in a list of important questions with corresponding answers. These were used as a type of requirement for the extension to be developed.

5.1 Brainstorming - Methodology

The first part of the design process was an internal brainstorming session consisting of two participants; the authors of this report. Despite the fact that the brainstorming session being very informal, it followed along the lines of Thomas Kelley's seven secrets for a better brainstorming (Kelley and Littman, 2001, pp. 54-56) to a certain extent.

One of the seven secrets is to have a clear goal with the session, which in this case was to produce several ideas regarding how to solve the issue. In order to understand the issue at hand, Kelley suggests that a very clear problem statement benefits the outcome of the session. Thus, three problem statements were created and used as foundation for the entire session:

- How can one visualize a discovery path in Qlik Sense?
- How can visualizations help the program to interact with the user?
- Which are the most important functionalities?

Another guideline for brainstorming sessions is to use catalysts for further inspiration. Catalysts are methods used to increase inspiration when the frequency of new ideas decreases. One way to do this is to build ideas on top of another (Rogers, Sharp, and Preece, 2011). The use of this type of catalyst was widely used during the brainstorming session.

For this session the approach of divide and conquer was used. This can mean two things; either that the session starts off with each group member having their own idea generating session and then those ideas are used as basis for the group discussion or that each member is responsible for a specific part of the problem at hand. This brainstorming session used the first alternative.

No suggestions were dismissed and one goal was to come up with as many solutions as possible. If the discussion faded on one solution a new approach was tried. During the process a whiteboard and papers were used to sketch ideas and listing possible solutions.

5.2 Brainstorming - Result

As mentioned above, the overall goal of the session was to come up with as many solutions as possible and to answer the mentioned questions. During the session several possible solutions were introduced and the questions were answered in the following way:

- How can one visualize a discovery path in Qlik Sense?
    - By some kind of tree representation where a selection sequence is represented by a branch
    - By a timeline where every selected value creates a new node
    - By a timeline where each selected field creates a new node (a node can contain many values from the same field)
- How can visualizations help the program to interact with the user?
    - By showing made selection when hovering over a node
- Which are the most important functionalities?
    - Users should be able to regret selections by button, mark and delete or by removing selection
    - Users should be able to name nodes by clicking on them and entering a name.
    - Users should be able to jump back in the discovery path by clicking on a node in the visualization
    - Users should be able to add text to the visualization

During the session new questions regarding the solutions arose:

- When can a selection be visualized?
    - When users confirm selections
    - When users click on a field value
    - When users click a save button (this would have to be implemented as part of the extension)
- How can one visualize removed selections?
    - By fading the colors of the removed ones
    - By color coding (e.g. red for removals and green for additions)
    - By removing the selection completely

All of these answers to the given questions were the result of many different solutions brought up during the session.

5.3 Brainstorming - Analysis

The goal of the brainstorming session, to come up with as many ideas as possible, enabled the creation of a solid foundation for the development of the lo-fi prototypes. The result from the brainstorming created a deeper picture of how the application could work, possible problems that could have occurred and clearer picture over how the design could have looked.

The brainstorm only had two participants. Thomas Kelley states that the participants in a brainstorm should have different backgrounds. He claims that it is better to mix someone from manufacturing, a customer and an expert rather than having 3 experts (Kelley and Littman, 2001). The reasoning behind this was that including experts, i.e. Qlik employees in the session would be very limiting as they already have developed their own picture of the program and how it is structured. Thus, their opinions and ideas would have been extremely biased towards certain solutions.

One could argue that there should have been participants without a technological background. However, the assignment involved taking into consideration the (at the time) current functionalities of Qlik Sense. Thus, a participant without any knowledge about the product would have had a difficult time generating ideas which fulfilled the needs of Qlik.

Looking back at this part it is clear to us that the session was performed by people with a suitable amount of knowledge within the areas of Qlik Sense and data visualizations. However, when taking a step back and looking at it objectively it is also clear that the session certainly would have benefitted from additional participants. That would have increased the amount of ideas and also the amount of approaches to certain problems.

As mentioned, additional questions arose during the session. These questions proved to be valuable at the end of this part as they helped in structuring the task at hand. The questions arose because things needed clarification. This goes to show the multiple fields of application of brainstorming; not only does it help generating ideas, it also helps developers detect issues so that they can deal with these issues.

5.4 Brainstorming - Conclusion

- Discovery paths in Qlik Sense can be visualized as a tree containing nodes.
- The most important functionalities of a visualization program are: removal of selections, naming of nodes, clickable nodes and addition of text to the visualization.

- A selection in Qlik Sense can be visualized when confirmation of selections has been made, a field value is clicked or a visualization button is pressed.
- Removed selections can be visualized with the help of faded colors, color coding or by complete removal from the visualization.
- Brainstorming sessions can benefit from consisting of more than two participants. However, groups consisting of two participants can still produce enough ideas to help development of products.

6 Iteration 1 – Lo-Fi prototyping & heuristic evaluation

This iteration describes the creation and result of the lo-fi prototypes. All prototypes are based on ideas from the brainstorm session.

6.1 Lo-fi prototype - Methodology

This iteration started off with a session where lo-fi prototypes were created. These prototypes were based on the ideas generated from the brainstorming session.

As mentioned before, lo-fi prototypes can take different forms. For this project two types were created; digital sketches showing different ways to include certain functionality and digital storyboards. Storyboards consists of a series of sketches showing how a user might progress through a task using the product under development (Rogers, Sharp, and Preece, 2011). The sketches and storyboards were created using Google's free drawing program, Google Drawings.

The first step of the creation of lo-fi-prototypes was sketching. After the sketches were completed a project manager gave his view of what was important for the resulting product. The evaluation revolved around what he considered important for the end users, the Qlik Sense customers, and how the developers at Qlik planned to include the resulting extension in their product.

The thoughts of the project manager were then used to decide what to keep in which sketches. Once this was decided new prototypes in the form of storyboards were created, based on the thoughts of the project manager.

After these prototypes had been created they were presented to one of the supervisors for this project. He then made a heuristic evaluation of the different prototypes. This was done through a Skype call with screen sharing.

6.2 Lo-fi prototype - Result

The lo-fi session resulted in many different sketches and prototypes, all of which fulfilled different desired requirements from the brainstorming session.

6.2.1 Sketches

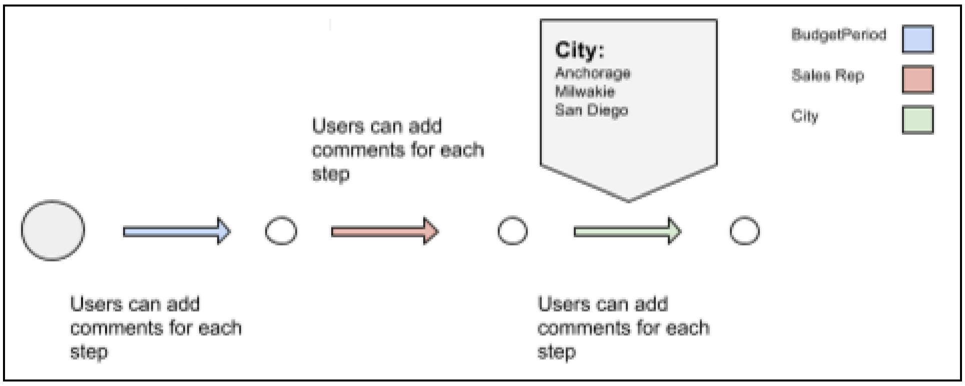

City:
Anchorage
Milwakie
San Diego
BudgetPeriod
Sales Rep
City
Users can add comments for each step
Users can add comments for each step
Users can add comments for each step

Figure 5: Sketch illustrating the basic functionality of the extension to be developed

6.2.1.1 Sketch 1 – Simple Selection

This prototype uses horizontal arrows to represent selections made by the user. Once a selection is made, an arrow is created and when the selections are confirmed a circle is created to visualize that the selection has been confirmed. The arrows are color coded depending on the selected field. It is possible to add comments to the visualization as well as hovering over an arrow to show all of the currently selected values from the field.

6.2.1.2 Sketch 2 – How to deselect?

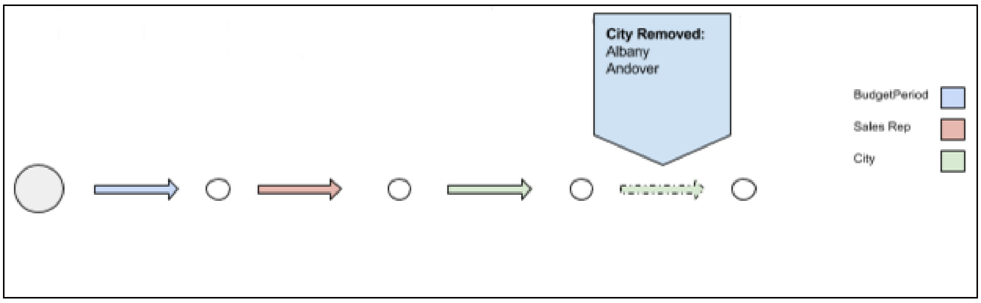


Figure 6: The first of the two suggestions of how to visualize deselections

As explained by its name, this sketch has one purpose; to elicit ways to handle deselections. It provides two different suggestions. The first handles deselection by appending a blurry arrow to the timeline (see Figure 6). Hovering over that arrow displays which deselections were made. This allows the user to see the complete selection history in the timeline.

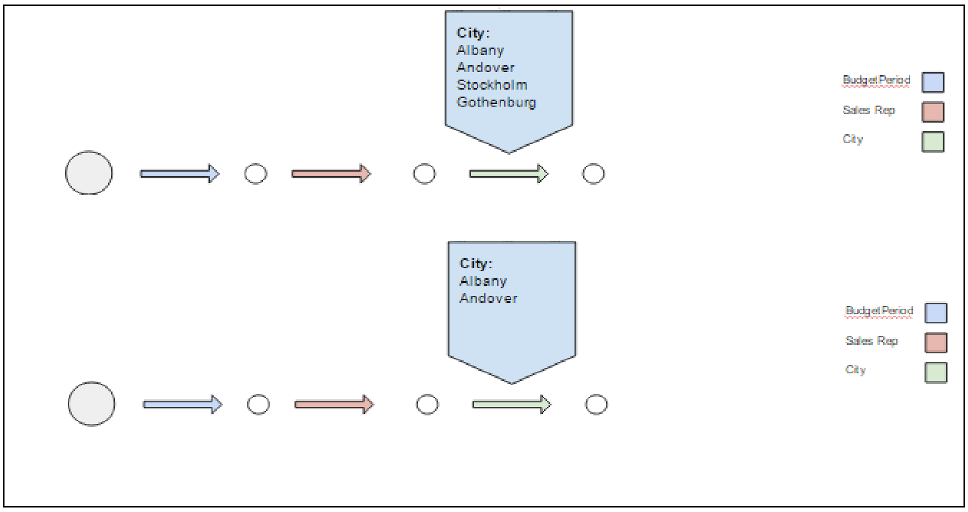


Figure 7: The second of the two suggestions of how to visualize deselections.

The second suggestion handles deselection by removing the values from the list of selections which appears when a user hovers over the corresponding arrow. In other words, no arrow is added if a user removes a value but the values in the corresponding field is changed (see Figure 7). The drawback with this method is that the visualization does not show deselections.

6.2.1.3 Sketch 3 – "Visualize"-button

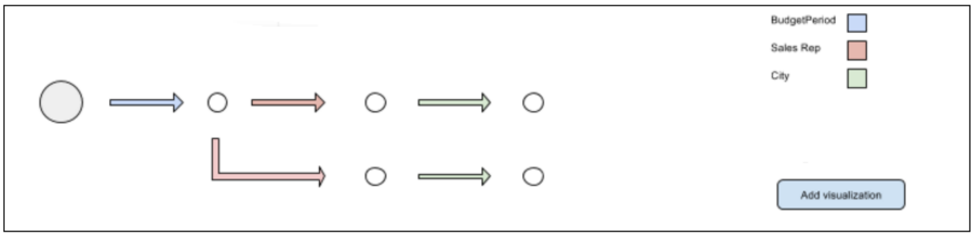


Figure 8: Sketch illustrating the tree made when saving a state

The third prototype contains a button which the user can press to visualize his/her current selection state. Once a selection state has been saved the user can make new selections, and save new states. New states are saved to the old which creates a tree showing the relationship between them (Figure 8).

6.2.1.4 Sketch 4 – Top to down

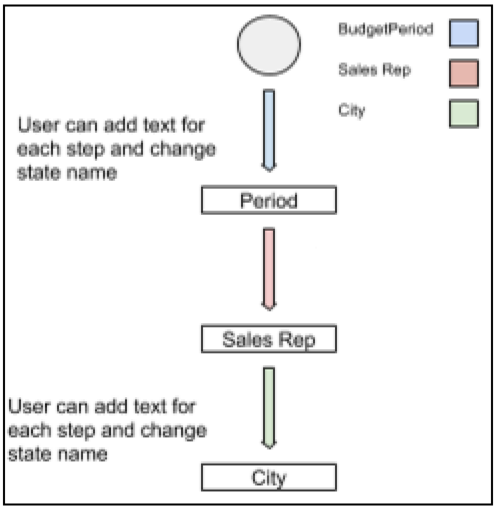

BudgetPeriod
Sales Rep
City
User can add text for each step and change state name
Period
Sales Rep
User can add text for each step and change state name
City

Figure 9: Sketch showing a different approach to the layout of the extension

This prototype was created in order to suggest a different way of presenting the visualization; by visualizing added selections from top to bottom (vertically). It gives two examples, one with color coded arrows (see Figure 9) and one without.

6.2.2 Project manager statement

The project manager made the following statements:

1. It is of utmost importance that selections are visualized in chronological order.
2. Every selection should be shown; even the ones which have been removed.
3. Enabling the user to interact with the visualization is important.
4. Scalability is important since the set of data can become very large.
5. Enabling the user to visualize the path to discovery in different ways is important.

6.2.3 Storyboards

As mentioned before, new prototypes in the form of storyboards were created based on the points above. All prototypes were created with the ultimate goal to fulfill as many of the five points as possible. Five storyboards were created in total; two which represented data discovery paths in a tree and three which represented it as a timeline.

6.2.3.1 Tree representations of data discovery paths

6.2.3.1.1 Prototype 1 – Selection order matters

This prototype visualizes data path discoveries in a tree. When a user makes selections and reaches an insight, that user can save the discovery path. When a discovery path is saved the selections are put into the tree in chronological order, starting from the center of the tree. This means that if a user selects A, then B, then C and then save that discovery path, this prototype would create three arrows: one from the root of the tree to A, one from A to B and one from B to C. If a user then were to clear all selections, then select B, then A, then C and lastly save the discovery path, that would result in two branches even if the data set of the two insights are exactly the same. This happens because selection order matters.

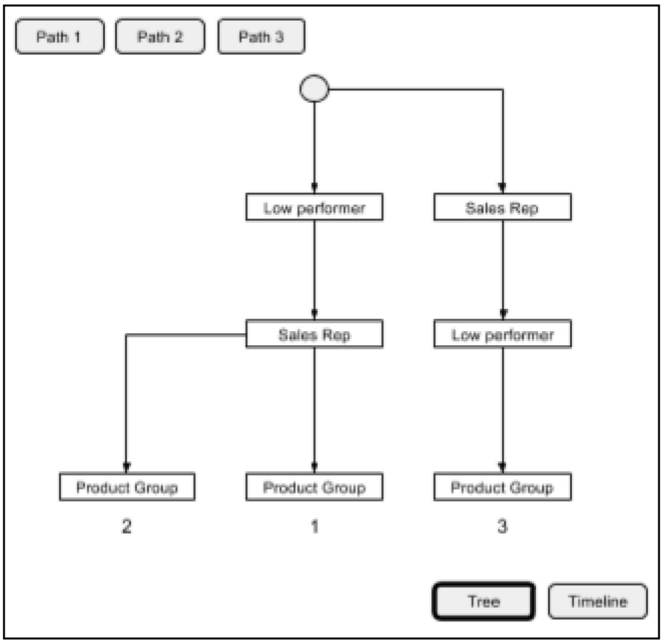


Figure 10: A screen capture of one suggested visualization prototype

Additionally, a button is created for each saved insight. Pressing this button will highlight the corresponding path to discovery in the tree. Thus, by pressing these buttons, the users can decide which insights to show and which to exclude. Note that in this tree representation of insights deselections are not displayed, only the set of selections which were selected when the user saved a discovery path is.

6.2.3.1.2 Prototype 2 – Selection order does not matter

This prototype has the same functionality as the first one but differs in one way; selection order does not matter in the tree representation of discovery paths. Given the example above with selections A, B and C, this prototype would consider the two discovery paths exactly the same since the order of selection does not matter. Therefor the tree would not be updated and still only contain one branch.

6.2.3.2 The history tab

The purpose of this tab is to enable the user to see and share every detail in how he/she made a data discovery. It shows selections and deselections in chronological order and it also shows data discovery paths. Three different prototypes were created, each with a different way of representing the same data.

6.2.3.2.1 Prototype 3 – Line of triangles

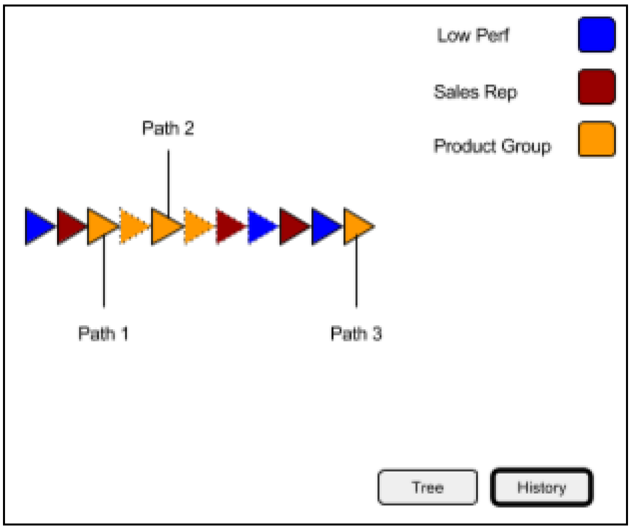


Figure 11: A screen capture of one suggested way to visualize selection history

This prototype visualizes each selection with a color coded triangle and each deselection with a faded triangle of the same color. Every selection and deselection is visualized from the moment when the user open the Qlik Sense application. When a user saves a discovery path the latest selection or deselection is marked with a tag so the user can browse the timeline for each saved selection.

6.2.3.2.2 Prototype 4 – Line of plus and minus signs

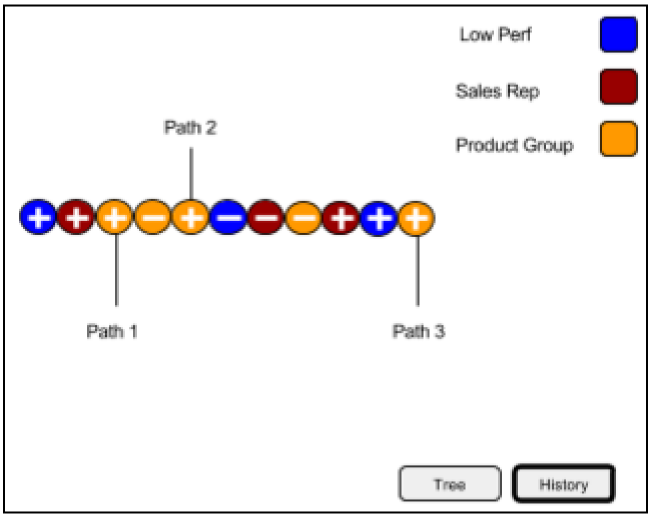
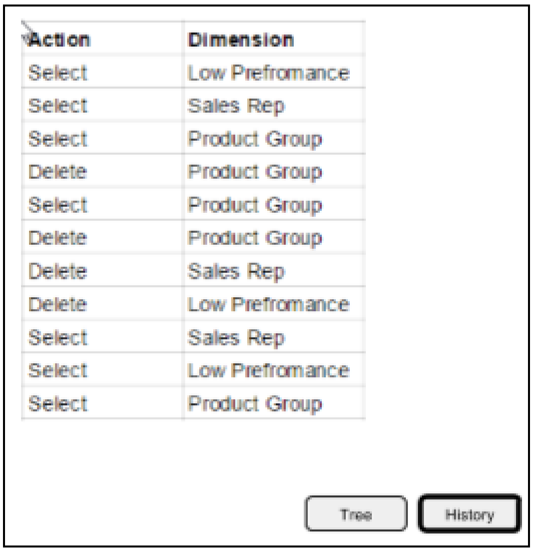


Figure 12: A second suggested way of visualizing selection history

This prototype has the same functionality as the previous one. However, it visualizes selections and deselections differently. Selections are visualized as color coded circles with plus signs in the middle, while deselections have minus signs in the middle of the circles.

6.2.3.2.3 Prototype 5 – Table



Action	Dimension
Select	Low Prefromance
Select	Sales Rep
Select	Product Group
Delete	Product Group
Select	Product Group
Delete	Product Group
Delete	Sales Rep
Delete	Low Prefromance
Select	Sales Rep
Select	Low Prefromance
Select	Product Group

Figure 13: A third and last suggestion where selections are represented in a table

This prototype uses a similar but much simpler approach. It uses a table where each row represents a selection or deselection. Each row also contains information regarding which field the action was performed on.

6.2.4 Heuristic evaluation of prototypes

This was done by the Qlik based supervisor for this project. During this session the five prototypes were introduced and evaluated individually. In order to make the feedback easier to grasp a list was created.

- Along the lines of what the project manager said, the supervisor also stated that selection order matters. Thus, the prototype *Selection order does not matter*, was not considered suitable for further development.
- He did not like the approach to choose one of the prototypes, instead he stated that the extension could include all solutions which the different prototypes suggested.
- The supervisor confirmed the previous mentioned usage areas for this extension; data presentation and data analysis. The first refers to when a user is presenting discoveries and the latter refers to when a user is in the process of making discoveries. Thus he suggested that both the tree and history representation of the selection data should allow the user to choose between showing deselections or not.
- He pointed out that being able to edit the tree and/or timeline must be included in the final solution since Qlik products are required to be very agile.
- He recommended that the final product included an activation button as well as a deactivation button. The first would be used provide the user with the ability to choose when to start visualizing discovery paths and the latter would be used to stop recording.

6.3 Lo-fi prototype - Analysis

6.3.1 Evaluations

As mentioned in the chapter *Theoretical background*, a heuristic evaluation should be done by usability experts or by double expertise people; in both usability and the product domain. Neither the project manager nor the thesis supervisor are usability experts. However, they are experts within the domain of the product and thus they were considered suitable to make the evaluations. Additionally, the project manager frequently interacts with Qlik Sense users, making him even more suitable for the evaluation since he has a good view of what the users generally want.

6.3.2 Feedback from project manager

The five points which the project manager brought up made the vision of the final product clearer. Using these points as basis, certain parts of the sketches could be removed.

The first point, regarding importance of chronology, initially meant that every selection made had to be represented by a new arrow, thus limiting further development to this approach. However, after the feedback from the supervisor it was clear that the solution could not consist of a single type of visualization. Instead the focus had to be on making the solution flexible. This ultimately led us to the conclusion to provide the user with option to include deselections or not.

The fourth point regarding scalability forced a new way of thinking when creating prototypes by combining this point with the first. If every selection is to be visualized then scalability might become an issue because of the large visualization.

6.3.3 Storyboards

The storyboards were a great asset in the process of eliciting functional requirements and helped in creating a clearer picture of what was needed in the final product. Up until the point when the storyboards were complete the ideas of the authors could not be explained in a proper and structured way. However, using the storyboards as a way of presenting ideas of appearance was not very helpful; for example, the three timeline prototypes.

Additionally, instead of using storyboards as a way to present a visualization suggestion, the functionalities of the timeline could have been suggested in one prototype while the style or appearance of the timeline could have been presented as a one-page drawing.

6.4 Lo-fi prototype - Conclusion

A product which visualizes discovery paths in Qlik Sense should:

- provide the user with several options of how to visualize the paths to discovery because different areas of application benefits from different types of visualizations.
- be developed in a way which allows the user to edit the visualization.
- enable users to decide when their selection history is stored.

7 Iteration 2 – Development of initial Hi-Fi prototype and user testing

This section describes the process of the development and testing of the hi-fi prototype. The hi-fi prototype was web-based and was created from the feedback given on the lo-fi prototypes.

7.1 Hi-Fi prototype

7.1.1 Hi-fi prototype - Methodology

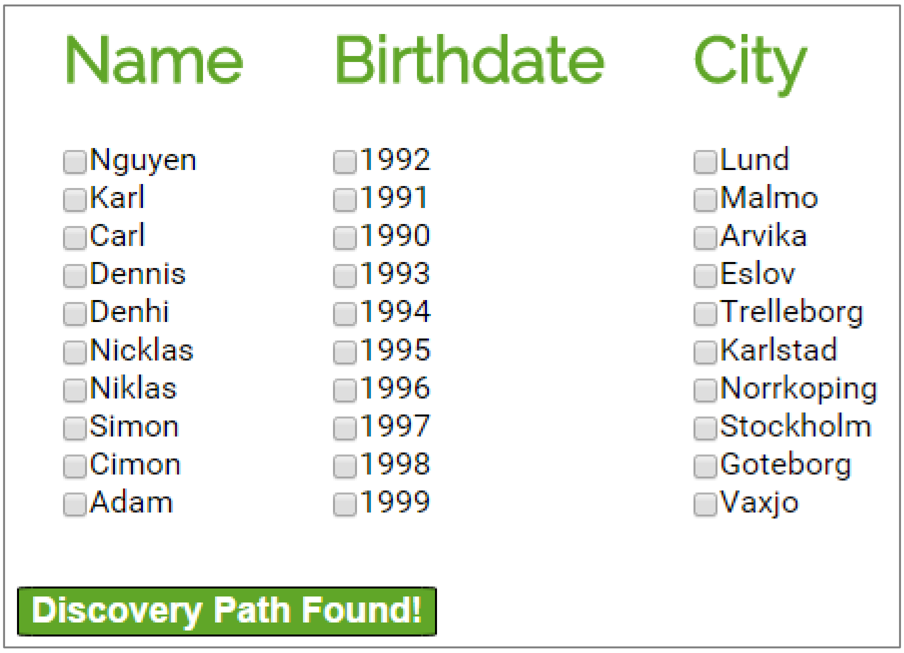
The initial hi-fi-prototype was based on the feedback which was received in the previous iteration. It was decided that this prototype would be created completely independent of Qlik Sense. This was done by making it web based which meant that certain functionalities included in Qlik Sense had to be simulated in the hi-fi-prototype.

The Javascript framework Data Driven Document (D3.js) was used to visualize the discovery paths.

7.1.2 Hi-fi prototype - Result

The first part of the system which had to be simulated was selections. To achieve this, HTML-checkboxes were used (see Figure 14). Checking a checkbox represented making a selection while unchecking it represented a deselection.

Once selections were simulated, a button for saving discovery paths was added. Upon pressing this button the user tells the system that he/she found a discovery path.



Name	Birthdate	City
□Nguyen	□1992	□Lund
□Karl	□1991	□Malmo
□Carl	□1990	□Arvika
□Dennis	□1993	□Eslov
□Denhi	□1994	□Trelleborg
□Nicklas	□1995	□Karlstad
□Niklas	□1996	□Norrkoping
□Simon	□1997	□Stockholm
□Cimon	□1998	□Goteborg
□Adam	□1999	□Vaxjo

Figure 14: A screen capture of the hi-fi-prototype's way of mimicking Qlik Sense's selection feature

Subsequently, the tree representation of discovery paths was developed (see Figure 15). At this stage of development not much effort had been put into the style and looks of the system. In order to make it look more appealing colors and the overall theme of the system were changed (see Figure 16).

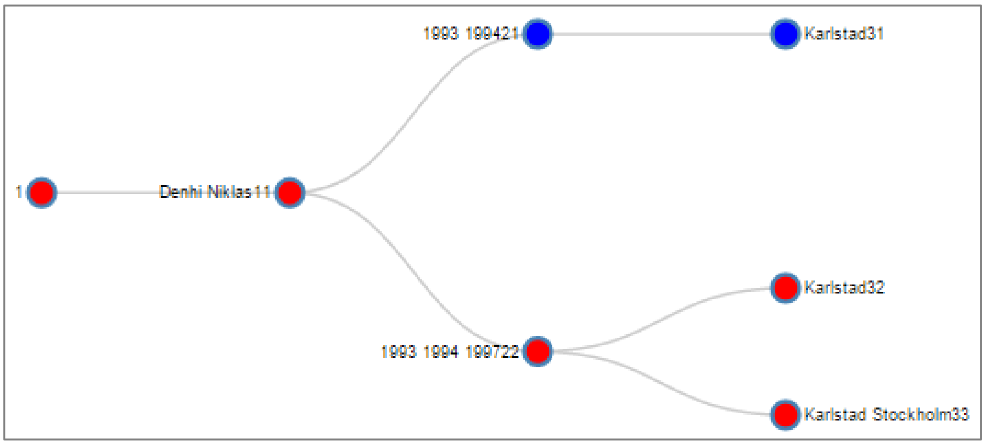


Figure 15: A picture of the path tree in early development

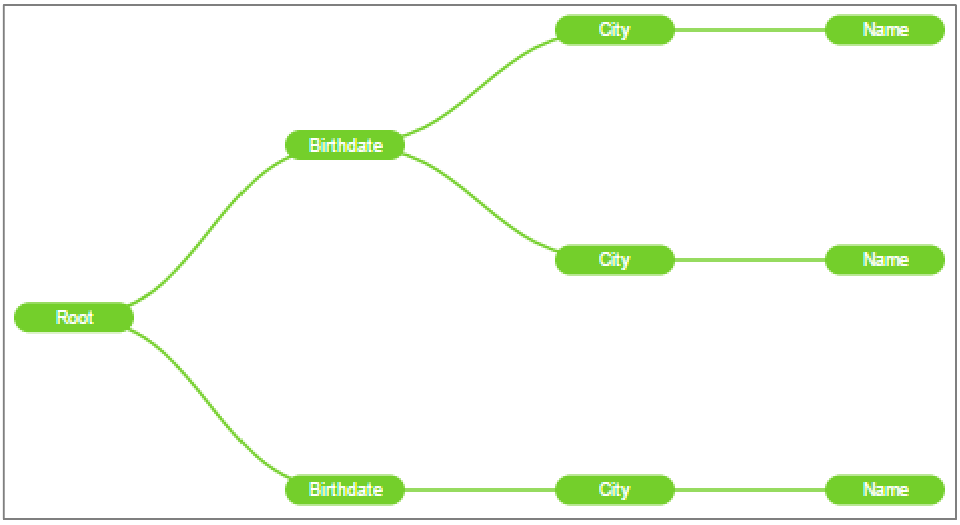


Figure 16: A picture of the path tree after styling had gone through a styling

Other than simply visualizing the paths the system was required to be interactive. Thus, three features were added:

- When a user hovers over a node in the tree, an information box is displayed containing all the selections of that path.
- When a user clicks a node in the tree, the selections of that path are selected.
- Users are able to choose which paths are displayed. Discovery paths which are not displayed are faded.

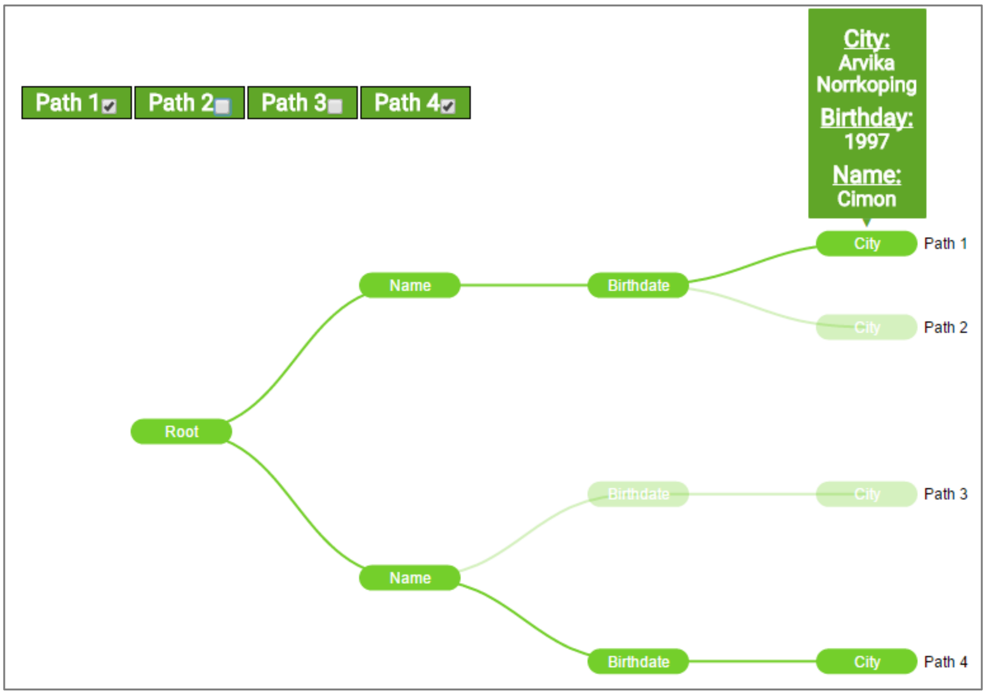


Figure 17: A picture of the visualization part of the hi-fi-prototype showing the tool tip functionality

Lastly, the requirement that the user should be able to view selections in chronological order was dealt with by developing a timeline which was based on the concept of the lo-fi-prototypes. Once the timeline was complete, two buttons were added, enabling the user to switch between tree- and timeline- visualization.

The final version of the hi-fi prototype is found in appendix A.

7.1.3 Hi-fi prototype - Analysis

7.1.3.1 Independent prototype

The decision to develop a prototype completely independent of Qlik Sense turned out extremely valuable in the end. It enabled focusing on the features of the prototype rather than technical details of Qlik Sense.

7.1.3.2 The final result and requirements

As previously mentioned, the base for this development process were the requirements elicited from the evaluations of the project manager and the supervisor:

- Display selections in chronological order – The final prototype provides the user with a completely chronological timeline and tree representation of selections. Hence, this requirement was fulfilled.
- Display deselections – Deselections are displayed in the timeline, not in the tree. The reasoning being that the tree would end up much cluttered. Because deselections are displayed in one of the modes, this requirement was considered fulfilled.
- Flexibility – Although the prototype provides flexibility to an extent, it is not one of the prototype’s most distinguished properties. However, before providing more flexibility to the user, the features developed at that time had to be scrutinized.
- Scalability – The prototype handles a greater amount of data in a scalable manner; both in the timeline, by letting the user scroll, and in the tree, by automatically restructuring the tree depending on the current amount of saved discovery paths.
- Interactive – The prototype handles user interaction is several ways. The user can hover over nodes to display information, select which discovery paths to display and which to remove and lastly make selections of a certain discovery path by clicking on its node.

The goal of the prototype was to hand out user tests on it. This was done by fulfilling all of the above requirements.

7.2 User testing

7.2.1 User testing - Procedure

The main goal of the user testing was to:

- Answer whether the users understood how to make use of the different functionalities of the prototype
- Answer whether the users found the prototype helpful in overviewing their previous selections and/or their paths to discovery

To investigate this a task description, namely a scenario, was created. The goal for the scenario was to make it as realistic as possible, thus putting the user in a context where he/she had actual use of the prototype. The scenario put the user in the role of a project manager whose boss had requested an evaluation of a certain data set. To perform this task the user was asked to use the prototype. Subsequently, the user was asked to perform certain tasks, all of which aimed to see how the user used the prototype.

Once the development of the scenario was completed a survey was produced in order to provide the testers with the opportunity to give an anonymous evaluation of the system. This survey contained both quantitative and qualitative questions.

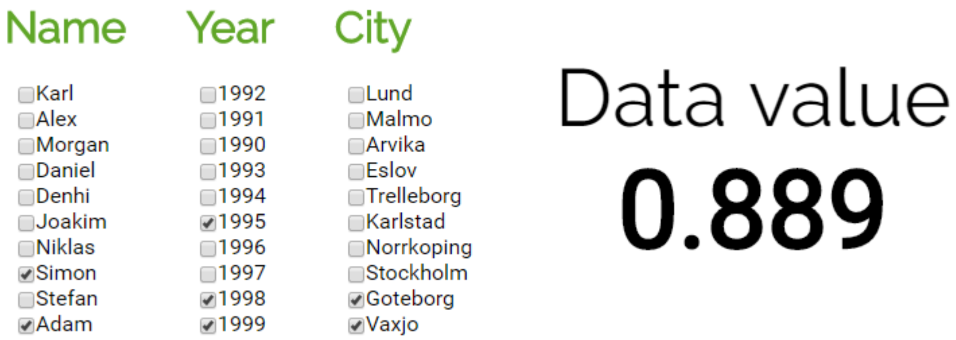
The scenario and the survey were both tested in a pilot study. The goal for this study was to detect any uncertainties in the scenario and/or the survey. In appendix B the scenario and survey used for testing is shown.

Because of no access to Qlik Sense costumers the test was performed on people with little or no previous experience with Qlik Sense. In total the test was performed on 10 persons with different backgrounds, ages and gender. The only requirement was that the testers knew how to use a computer.

7.2.2 User testing - Result

7.2.2.1 Pilot study

The pilot study was handed out on two people. The result of the pilot study was that redefinitions of certain questions had to be done and that the users were in need of some context regarding the checkboxes; to make them useful in some way. Thus it was decided to make different combinations of checkboxes produce a value depending on which ones the user had chosen (see Figure 18).



Name

☐Karl
☐Alex
☐Morgan
☐Daniel
☐Denhi
☐Joakim
☐Niklas
☑Simon
☐Stefan
☑Adam

Year

☐1992
☐1991
☐1990
☐1993
☐1994
☑1995
☐1996
☐1997
☑1998
☑1999

City

☐Lund
☐Malmo
☐Arvika
☐Eslov
☐Trelleborg
☐Karlstad
☐Norrkoping
☐Stockholm
☑Goteborg
☑Vaxjo

Data value

0.889

Figure 18: A picture of the selection functionality of the hi-fi-prototype after the data value functionality was added

7.2.2.2 Assignments

Assignment 1: Determine which names, years and cities a path contains.

Statement: Assignment 1 was easily solved

Result:

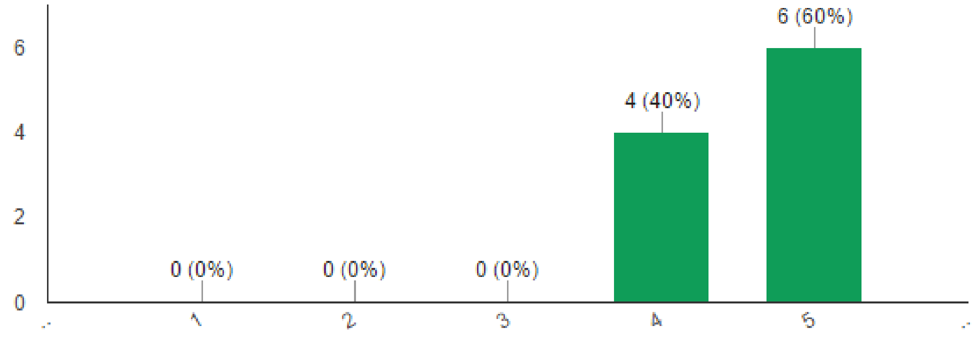


Figure 19: Result of user testing from assignment 1.

Assignment 2: In a path, determine which was selected first: name, year or city

Statement: Assignment 2 was easily solved.

Result:

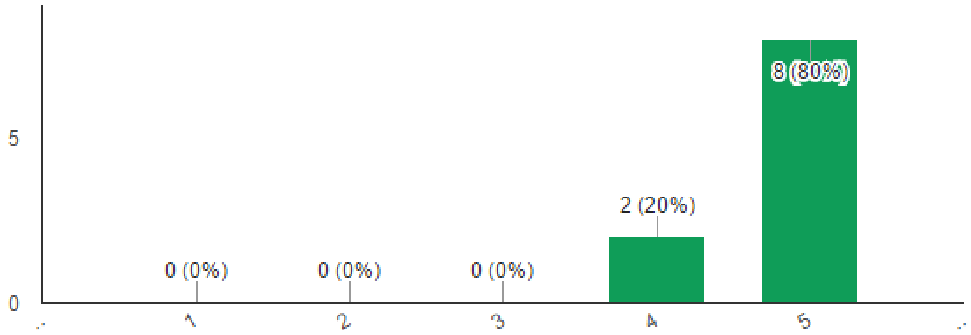


Figure 20: Result of user testing from assignment 2.

Assignment 3: Determine which selections and deselections were made to reach a path.

Statement: Assignment 3 was easily solved.

Result:

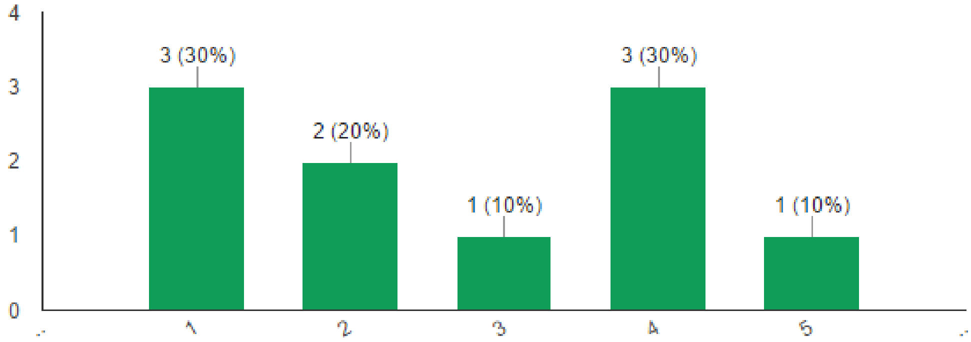


Figure 21: Result of user testing from assignment 3.

Assignment 4: Without clicking the checkboxes, determine the Data Value of a path.

Statement: Assignment 4 was easily solved.

Result:

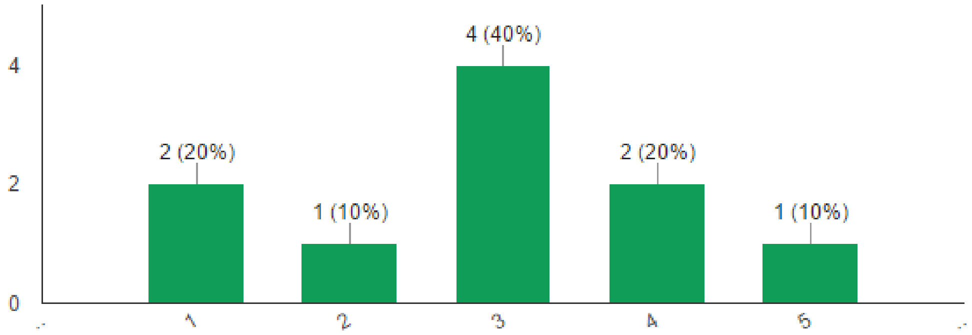


Figure 22: Result of user testing from assignment 4.

Assignment 5: Remove a path.

Statement: Assignment 5 was easily solved.

Result:

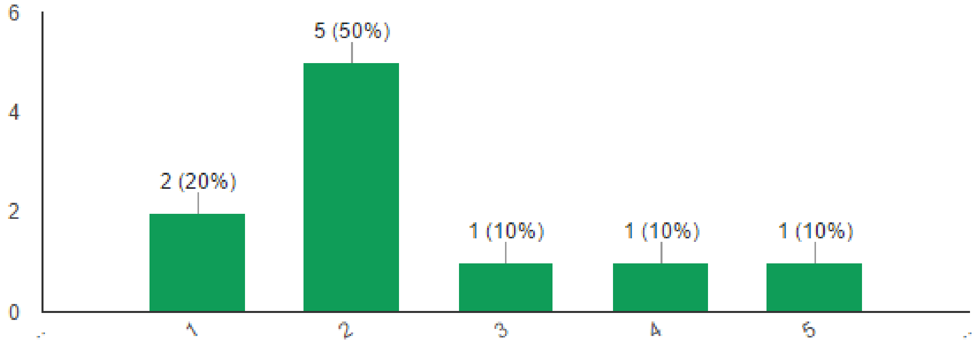


Figure 23: Result of user testing from assignment 5.

7.2.2.3 General impression of prototype

Statement 1: The prototype made it easier for me to overview my previous selections.

Result:

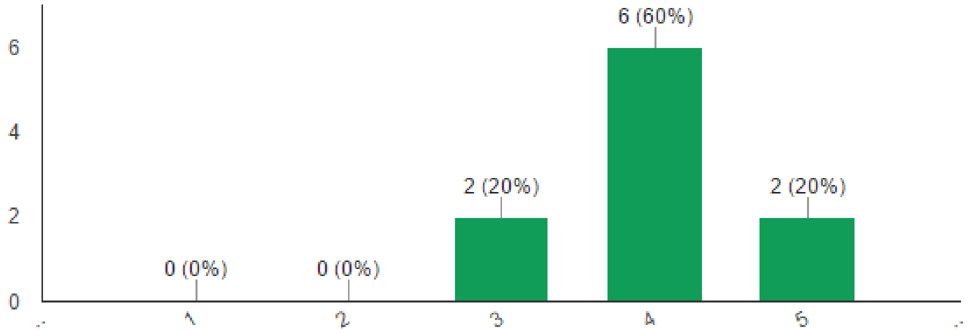


Figure 24: Result of user testing from statement 1.

Statement 2: The prototype is intuitive.

Result:

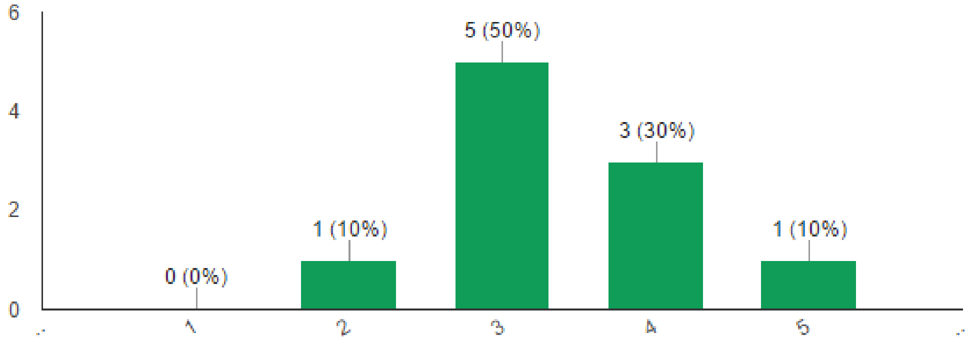


Figure 25: Result of user testing from statement 2.

Statement 3: The prototype is esthetically appealing.

Result:

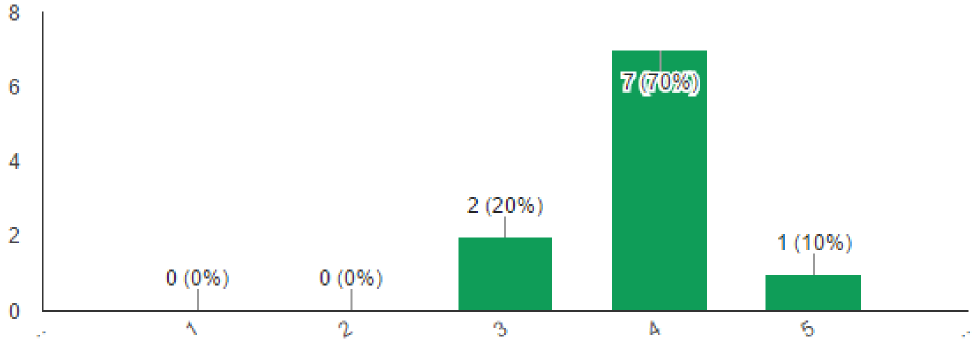


Figure 26: Result of user testing from statement 3.

7.2.2.4 Qualitative questions

What is your overall impression of the prototype?

After summarizing the results from this question the following points were brought up:

- The prototype had parts which were very well thought through:
    - The tree
    - The tooltip pop-up
- The prototype had parts which should have been developed in a different way:
    - Path deletion
    - Timeline
- The usage area of the prototype seemed limited

Do you have any suggestion for additional functionality?

The following suggestions were brought forth by the testers:

- Instead of having two buttons for switching between tree and timeline views, a single button could be used. The text on the button should always be the opposite of the current mode. Thus, adding constraints to the users.
- An undo-button so users can correct their mistakes, such as deleting the wrong path.
- Confirmation box when users are about to delete a path.

7.2.3 User testing - Analysis

7.2.3.1 Pilot study

As mentioned above, the pilot study concluded in a revision of the user tests. Before the study was handed out, the plan was to give the testers a set of predefined selections and discovery paths to work with. The problem with this, which was discovered during the pilot study, was that the testers had problems in understanding why they made these selections. The whole point of the project is to give users a way to visualize selections which they find important so if they are not aware of why the selections are important, the test loses its purpose.

By introducing and implementing the *Data Value*-concept into the product, the testers could now understand why a certain combination of selections was interesting, namely because the data value was higher than 0.6. This also mirrored the discovery process in Qlik Sense in a better way since the users could make selections freely during the test and not being forced into certain combinations.

7.2.3.2 The user tests

Deriving the results from the user tests and the surveys it was quite obvious which functionalities and parts of the prototype needed prioritization during future development.

The testers had difficulties in using the timeline. Additionally, they did not understand the purpose of it either. This result was not unexpected as the tree had been prioritized over the timeline during development. However, it was unexpected that the results were so poor. A large amount of time had been put into the timeline and to see that the testers found it close to useless was very surprising. It just goes to show how blinded developers can be by their own creations. In the end, it was very clear that the timeline was one of the major issues with the prototype at this point of time.

The other major part of the prototype which caused difficulties during the tests was the deletion of saved discovery paths. This was implemented in a way where the users were to select which paths to delete and then press a delete-button. However, during the tests, it was discovered that most users found the opposite way of doing it most intuitive; namely to press *Delete* and then to select which paths to delete. As the results were so clear regarding this matter it was decided to change the deletion functionality to match the way that almost all testers performed the task.

7.2.3.3 Qualitative together with quantitative data

The approach with qualitative and quantitative questions was a great asset and the result allowed for more detailed information regarding the parts of the prototype which received poor results. Even if the qualitative data was used as the primary data source, the qualitative data put words on those results.

7.2.4 User testing - Conclusion

- Performing user tasks on pre-defined selections and discovery paths left two out of two pilot testers unaware why these selections were important.
- Adding functionality which allowed testers to freely make their own selections and search for discoveries themselves put them in a more realistic context during testing.
- Visualizing selections and deselections as a straight line of nodes confused half of the test subjects rather than helping them.
- Deleting discovery paths by first marking a path and then pressing a delete-button made four out of ten test subjects unintentionally remove paths which they did not intend to remove.

8 Iteration 3 – Final Product Implementation & Sense Integration

This was the longest iteration and the phase where the final product was implemented as an extension in Qlik Sense. The design shown below was created based on the result from user testing.

8.1 Final product - Methodology

During this phase and iteration the product was adjusted according to the feedback from the user testing. Once this was done it was integrated into Qlik Sense in order to put it into its real context. Once integrated, the product was reviewed by the developers and the project supervisor and after several tweaks and adjustments to the final product arose.

8.2 Final product - Result

The final product enables the user to store and visualize his/her path to discovery simply by pressing a button once he/she found an interesting set of data (marked in green in Figure 27). When this button is pressed the user is asked to enter a name for the path which he/she decided to save. This name is from then on displayed in the visualization.

The product also provides the user with the option to visualize exactly which actions that he/she performed within Qlik Sense from the moment that the program was opened. This visualization mode is called History. Both of the above data sets are visualized as trees.

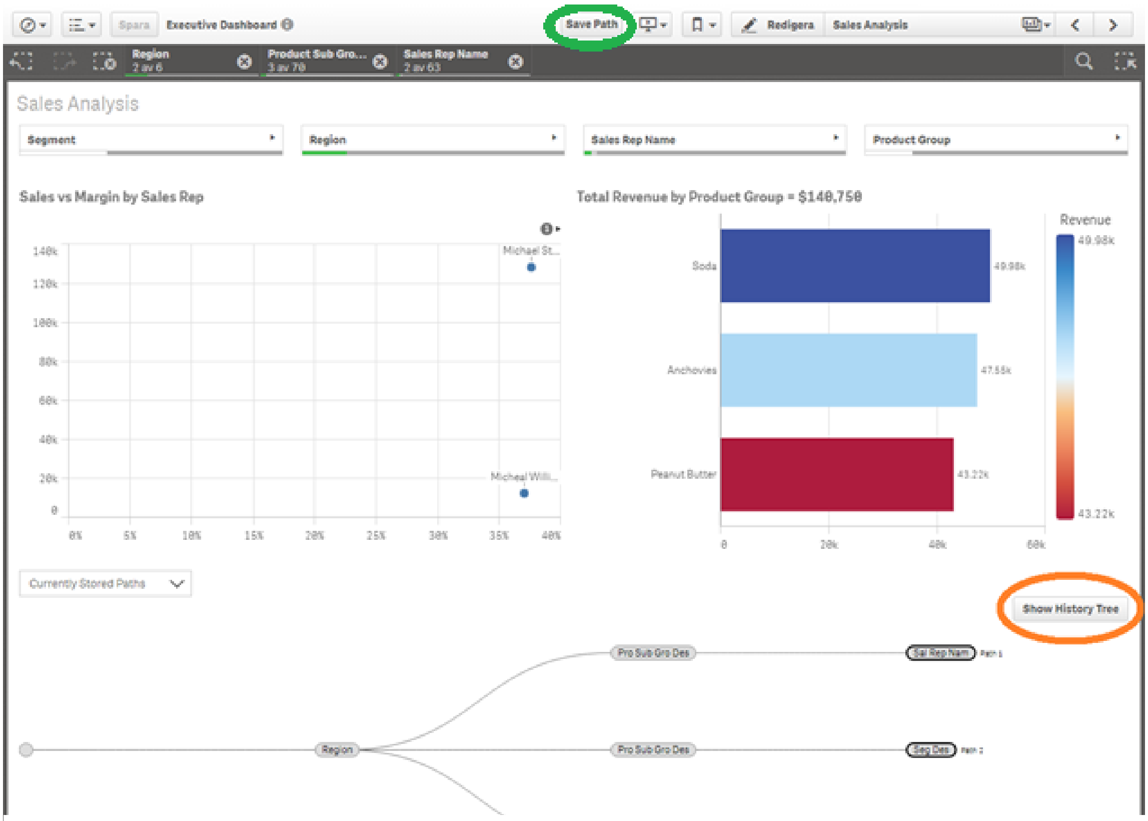


Figure 27: A picture of Qlik Sense after the extension had been integrated. The green circle shows the button that the user presses to save paths. The orange circle shows the button which allows the user to toggle between the two modes.

The product provides the user with the option to switch between the two modes whenever he/she desires; also with a simple click of a button (marked in orange in Figure 27).

Additionally, the product has a feature which enables the user to choose which of the previously stored paths he/she wants to display and/or remove. The list of stored paths is contained in a drop-down list (marked in blue in Figure 28). If a user unmarks a path in this list, the product will remove that path from the visualization but still keep the data stored (illustrated in Figure 29). However, if the user chooses to delete the path by pressing the corresponding trash bin, the path data as well as the visualization are cleared.

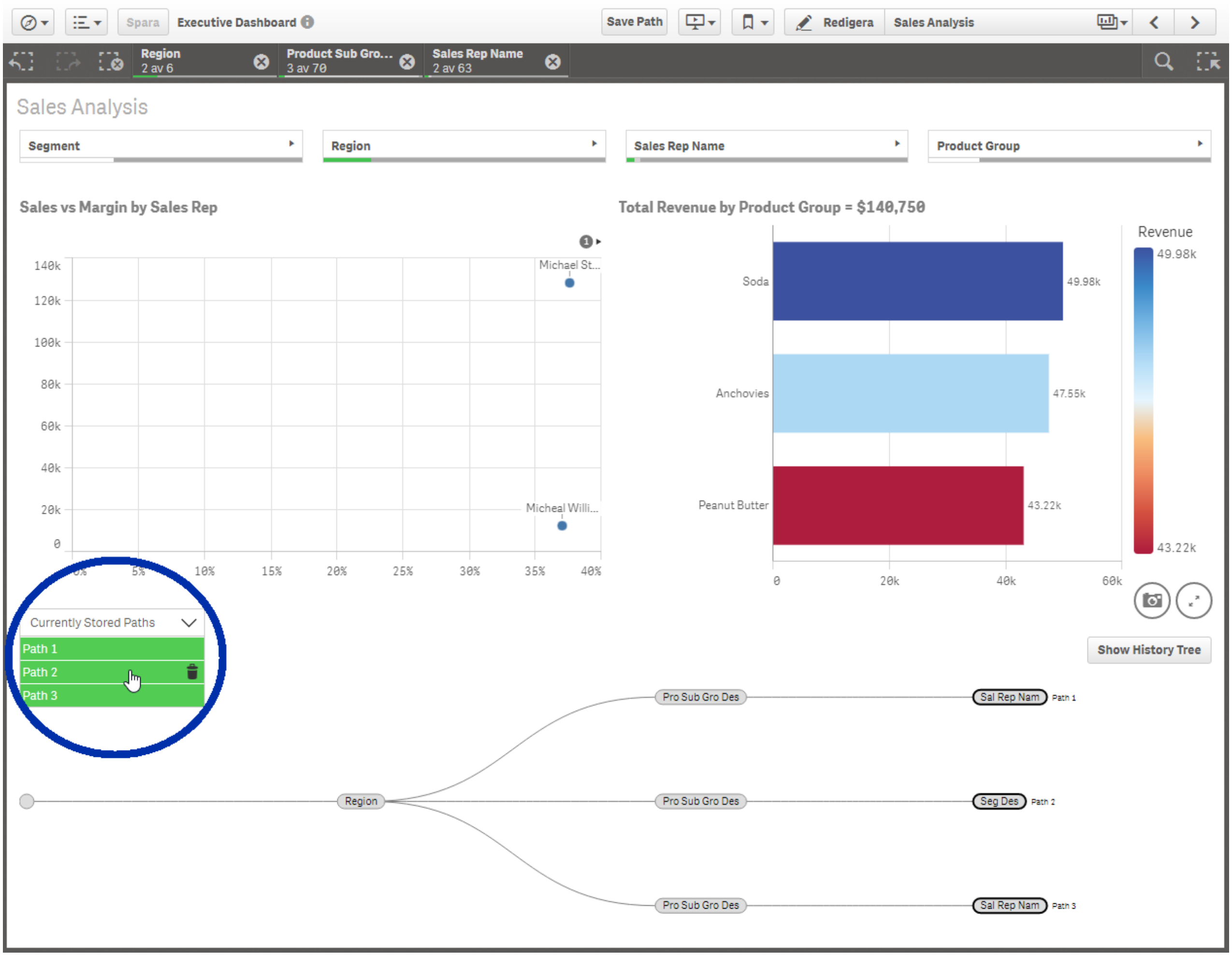


Figure 28: A picture which shows the drop-down menu containing all the currently stored paths.

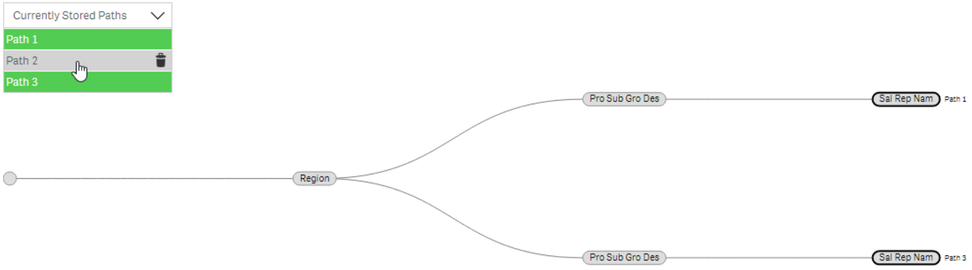


Figure 29: A picture showing how a user can hide paths by unmarking them in the drop-down menu.

The final product enables the user to view the content of nodes by hovering over them. A tooltip containing the data of the node will appear only if the node contains more than one selection. This is illustrated in Figure 30.

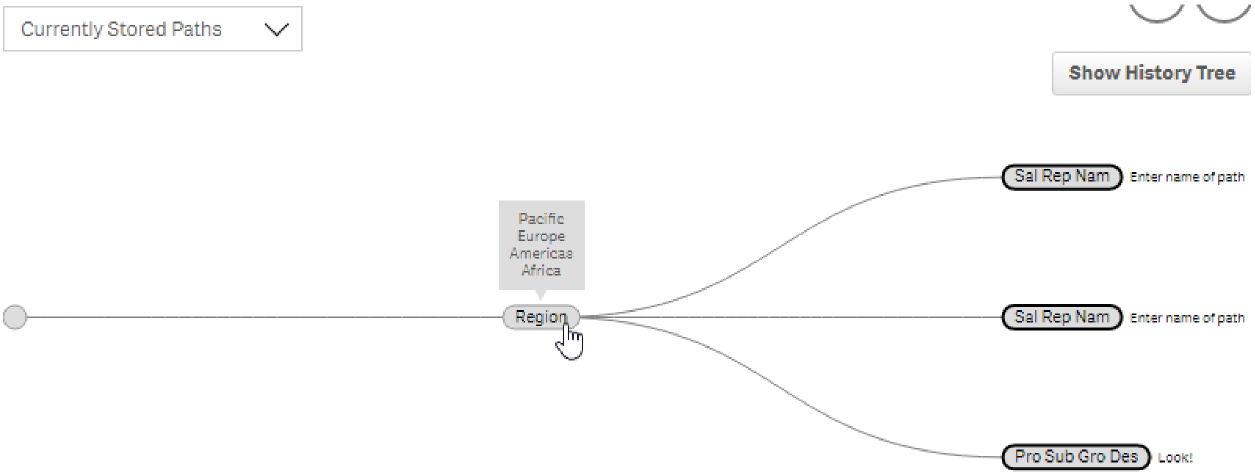


Figure 30: A picture showing the final version of the tooltip functionality.

8.3 Final product - Analysis

8.3.1 Updates during iteration 3

Integrating the result of the hi-fi-prototyping and user testing into Qlik Sense brought along multiple new design and implementation problems.

8.3.1.1 Colors

As seen in the pictures of the hi-fi-prototype, the color theme for the product was green. The reasoning behind this was that Qlik uses that type of green for their logo among other things and thus it was assumed that such a color scheme would blend in with the rest of the product. There was also positive feedback from user tests about the esthetics of the product.

However, once it had been integrated into Qlik Sense it was clear that the logo-green was not used at all within the product. Instead, the product used a black and white color scheme for most of its components and only colors as a tool to represent a range of data values. For example, if a data value was above a certain threshold it would be represented with one color and if it was below that threshold it would be represented with another color.

The product which was created during this project did not contain data within certain ranges and thus it was decided that the color scheme for it would be black and white, to match as much of the rest of Qlik Sense as possible.

8.3.1.2 Timeline revamp

The user tests clearly suggested that the timeline mode of the hi-fi-prototype did not accomplish its goal; to make it easier for users to overview every single selection that they made since the product was initiated. To deal with this issue a complete revamp of that functionality was made.

Instead of the previous solution, where selections in Qlik Sense were represented on a straight line of nodes, it was decided, together with the project supervisor, that the new approach would go along the lines of the tree-like solution that had been used for the path tree. This increased the cohesion of the product; tree-like visualizations for both modes. It also deals with the issue that a single line of nodes is not very scalable nor is it easily overviewed. The old solution contained several other issues. For example, when a user is browsing through data it is very common that he/she makes a selection only to deselect it again. By using a visualization which visualizes this in a minimalistic way, scalability is achieved.

As mentioned above, this mode is now called history and is easily switched to by pressing a button. Because that visualization is now much more similar to that of the path tree, the user is put in a position where he/she feels that the history mode is much closer at hand compared to the previous solution. Instead of having to grasp a completely different visualization the user can now simply toggle between two visualizations which only differs in the level of detail, and not in the layout.

8.3.1.3 Path drop-down list

As seen above, the hi-fi-prototype appended a button to itself for each discovery path that the user saved. This worked well in the context of a standalone product. However, as the prototype was integrated into Qlik Sense it was obvious that this was not a very scalable solution. This was an issue which unfortunately was overseen because we chose to perform user tests with a small amount of discovery paths and also on a standalone solution.

Even if the issue had been overlooked during the user testing we quickly realized that it had to be solved. In order to do this each project member had to come up with a solution. The process ended up in a decision to implement a drop-down list for all saved paths. Not only is this solution more esthetically pleasing, it also solves the scalability problem which appending elements for each saved path causes.

Additionally, we decided to add the delete functionality to this drop down menu. As seen above, the delete button is represented by a trash bin. This is a commonly used metaphor and makes the deletion of paths much more intuitive compared to a separate delete button.

8.3.1.4 Minimalistic tooltip

As seen in the pictures of the hi-fi-prototype, the user can hover over a node in the tree to see the selected values. This functionality remains in the final product but has been slimmed down to display only when it is essential for the user. The reasoning behind this was that tooltips are there to show something important and when a program shows a large tooltip for every single node, even if the information is already located elsewhere, the user runs the risk of not paying attention to future tooltips which contain important information.

8.3.1.5 Transitions

The hi-fi-prototype did not use animations. This could cause confusion when changes were made to the visualized tree. Nodes would move from their old to their new position in milliseconds.

To handle this we used transitions. What these do is quite self-explanatory. Say there is a node *N* and two positions *A* and *B*. Node *N* moves from position *A* to position *B*. This can be visualized as instant, which it was in the hi-fi-prototype, but making the node move over some time period lets the user follow how the node moved. For small amounts of data one can argue that transitions are unnecessary. However, once the visualized data grows they make the program understandable.

9 Discussion

Thoughts and opinions on our design process and a list over future improvements on the final product.

9.1 Process

One of the biggest challenges for us has been to truly understand what we want to achieve with our product and how the users use Qlik Sense. What does the user want? How should this functionality work? Will the functionality benefit the user or just make the product more complex? Is this possible to implement in Qlik Sense as an extension? These are questions which we have asked ourselves many times during the project. It is not easy to implement and design an extension for a software when you initially are not aware of how the costumers use the software. To design an extension that should make it easier for the costumers to use the software is even more difficult. To do this we have tried to follow the user-centered design methodology as much as possible. Our process has been iterative to continuously receive feedback of the design, making it easy to notice and correct missteps in an early stage.

We created two prototypes before we started implementing the final product, thus giving us the opportunity to test the design before we invest time into implementation. The prototypes did not only give a better view of the final design, but also how the underlying data structures could be implemented.

With no access to real costumers we have not been able to follow user-centered design completely. Instead we have used potential customers and designers from Qlik for testing. Due to this limitation we decided to create our hi-fi prototype completely outside of Qlik Sense. By doing so it was easier for us to test our product on people who had no previous experience of Qlik Sense. Instead of explaining how Qlik Sense works and how it is used, all focus was centered on our product.

If we were involved in a similar project again, we would change certain parts of our approach. Firstly, we believe that we should have put some more time into the initial phase of the project to understand how the users use Qlik Sense. The meeting with the project manager from Qlik where we discussed how the costumers use the software took place too late.

Working in an iterative manner has many advantages but also its drawbacks. Working in short cycles where you test and implement features continuously were confusing at times. In some stages it felt like we lost track and did not have a clear picture of the final product. Another approach would be to put more time into planning and make a more detailed design prototype before starting with implementation. Whether this approach would have been better is hard to say. However, we believe that it could have saved us time.

9.2 Future improvements

Due to the time scope all planned functionalities of the final product could not be implemented. If one is to put additional time into the product, we suggest that the features below are prioritized.

9.2.1.1 Current location in tree

To make the visualization easier to understand feedback regarding where the user currently stands in the trees should be shown. When the user makes selections a branch is created and added to the tree when the path is saved. To visualize where the user stands in the tree this branch could be visualized as if it was added to the tree with a different color-coding on its nodes, but not saved to the tree until the user saves the path.

9.2.1.2 Delete in history tree

In the current version the user can not remove paths in the history tree, meaning that the history tree shows all the selections he/she has made since he/she started the application. If the history tree becomes large this is not a scalable solution. To solve this problem many approaches can be considered. One way is to have a record button that allows the user to choose when to save the selection history. Another way would be to allow the user to delete the branches in the tree.

9.2.1.3 Select path in history tree

In the current product the user cannot select paths in the history tree. To allow the user to select which path to be shown in the history tree would make the program more interactive and scalable.

9.2.1.4 Merge nodes in history tree

In the history tree, nodes contain one or many values depending on how the selection was made. Many nodes close to each other in a branch may therefore be from the same field. To show all nodes separately is sometimes what the user wants, but in some cases this makes the tree unnecessarily large. To make it possible for the user to merge nearby nodes would make the tree more scalable.

10 Conclusion

- To visualize a set of discovery paths as a tree fulfilled the requirements of Qlik employees. It was also well received during user testing.
- The disadvantages of visualizing discovery paths as a tree is that once the tree becomes too large, the user might lose his/her overview of the paths. This is solved by providing the user with an option to display only the set of paths currently chosen by him/her.
- A discovery path should contain information regarding which selections the user has made and in which order the user made these selections.
- The functionality to visualize discovery paths in Qlik Sense can both be implemented as an extension and as a part of the Qlik Sense core. Which of these options is best suited depends on the area of application. If one intends to use one visualization for each sheet, then an extension is a suitable approach. On the other hand, if one intends to use one visualization for each application, then the visualization is well suited as part of the Qlik Sense core.

11 References

Rubin, J., Chisnell, D. and Spool, J. (2008) Handbook of usability testing: How to plan, design, and conduct effective tests. 2nd edn. Indianapolis, IN: Wiley, John & Sons.

Schön, D.A. (1984). The reflective practitioner: How professionals think in action. Aldershot: Basic Books.

Gould, J. D., & Lewis, C. (1985). Designing for usability: Key principles and what designers think. Communications of the ACM, 28(3), 300-311.

Barnum, C. (2010) Usability testing essentials: Ready, set...Test. United States: Morgan Kaufmann Publishers.

Henningsen, D.D. and Henningsen, M.L.M. (2013) 'Generating ideas about the uses of Brainstorming: Reconsidering the losses and gains of Brainstorming groups relative to nominal groups', Southern Communication Journal, 78(1), pp. 42–55. doi: 10.1080/1041794x.2012.717684.

Wibeck, V. (2010). Fokusgrupper : om fokuserade gruppintervjuer som undersökningsmetod. Lund: studentlitteratur AB.

Pruitt, J. and Adlin, T. (2005) The persona lifecycle: Keeping people in mind throughout product design. San Francisco: Morgan Kaufmann Publishers, an imprint of Elsevier.

Carroll, J.M. (2000) Making use: Scenario-based design of human-computer interactions. United States: MIT Press.

Kelley, T. and Littman, J. (2001) The art of innovation: Lessons in creativity from IDEO, America's leading design firm. New York: Bantam Doubleday Dell Publishing Group.

Kaner, C. (2003) 'An introduction to Scenario Testing', United States : Florida Tech

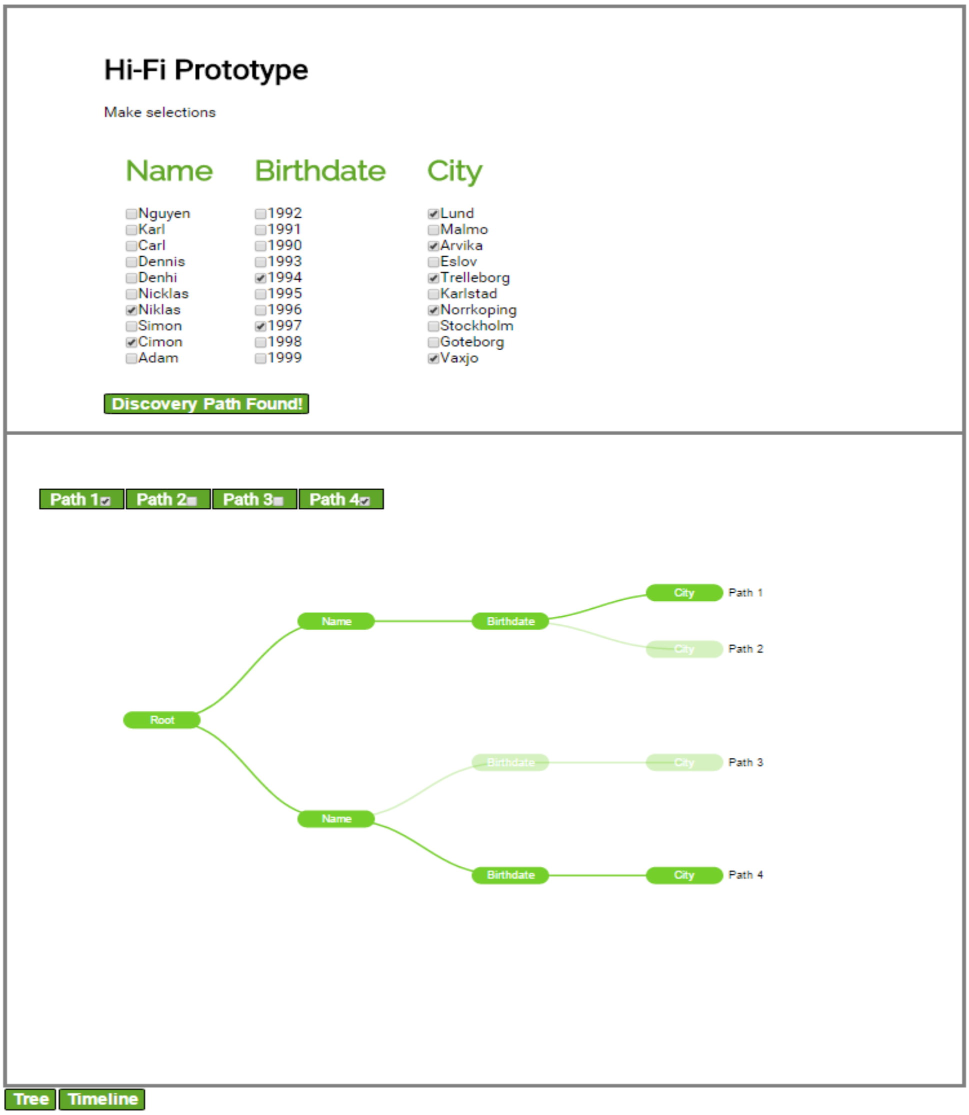
Qlik.com, 'Our history', Available at: http://www.qlik.com/company/history (Accessed: 01 February 2016)

Wills, G. (2008) Handbook of data visualization (Springer handbooks of computational statistics) (Springer handbooks of computational statistics). Edited by Chun-Houh

Chen, Wolfgang Hardle, and Antony Unwin. Berlin: Springer-Verlag Berlin and Heidelberg GmbH & Co. K.

Appendix A Hi-fi prototype

Figure over the web-based hi-fi prototype used for testing.



Hi-Fi Prototype

Make selections

Name

□Nguyen
□Karl
□Carl
□Dennis
□Denhi
□Nicklas
☑Niklas
□Simon
☑Cimon
□Adam

Birthdate

□1992
□1991
□1990
□1993
☑1994
□1995
□1996
☑1997
□1998
□1999

City

☑Lund
□Malmo
☑Arvika
□Eslov
☑Trelleborg
□Karlstad
☑Norrkoping
□Stockholm
□Goteborg
☑Vaxjo

Discovery Path Found!

Path 1☑ Path 2□ Path 3□ Path 4☑

Tree Timeline

Appendix B Usability Test

1.1 Test

Introduktion:

I detta test kommer du att få använda ett datafiltreringsprogram. Testet inleds med ett scenario i vilket du ska utföra en uppgift. Om du undrar någonting, fråga oss.

Scenario:

Du är projektledare för GoodBznz AB, ett företag som säljer kontorsutrustning.Du har fått data från din chef som han vill att du utvärderar.

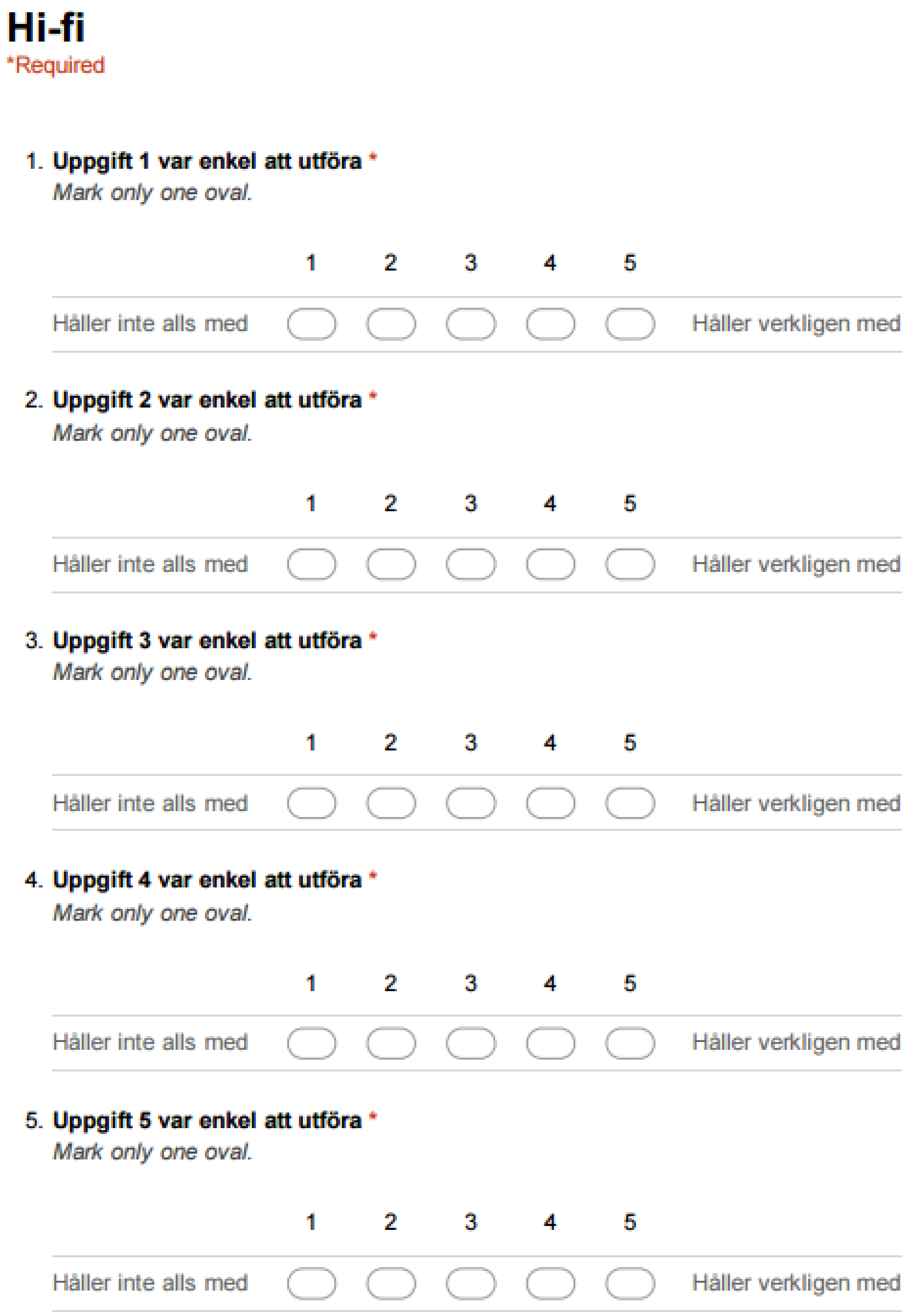
Till din hjälp har du programmet Insights. Detta program tar in data för samtliga säljare, orter och årtal. Insights är ett smart program som beräknar hur intressant datan är genom att ge användaren möjligheten att filtrera den. Detta görs med hjälp av checkboxar. När du filtrerar datan så beräknar Insights ett värde, som kallas *Data value*, mellan 0-1 där 0 är icke-intressant och 1 är mycket intressant. Din uppgift är att finna och spara undan 5 stycken intressanta (>0.6) datakombinationer (Discovery paths).

När du sparat dessa paths, fortsätt nedan:

Du ska nu hålla ett möte med din chef där du presenterar och hanterar datan. Med hjälp av Insights, gör följande:

1. Vilka namn, år, städer ingår i Path 4?
2. I Path 3, vilket valdes först: namn, år eller stad?
3. Vilka checkboxar markerade respektive avmarkerade du för att ta dig från Path 3 till Path 4?
4. Utan att röra checkboxarna, ta reda på *Data value* för Path 1
5. Chefen anser att Path 3 inte är intressant. Ta bort denna.

1.2 Survey



Hi-fi

*Required

1. **Uppgift 1 var enkel att utföra** *

Mark only one oval.

	1	2	3	4	5	
Håller inte alls med	○	○	○	○	○	Håller verkligen med

2. **Uppgift 2 var enkel att utföra** *

Mark only one oval.

	1	2	3	4	5	
Håller inte alls med	○	○	○	○	○	Håller verkligen med

3. **Uppgift 3 var enkel att utföra** *

Mark only one oval.

	1	2	3	4	5	
Håller inte alls med	○	○	○	○	○	Håller verkligen med

4. **Uppgift 4 var enkel att utföra** *

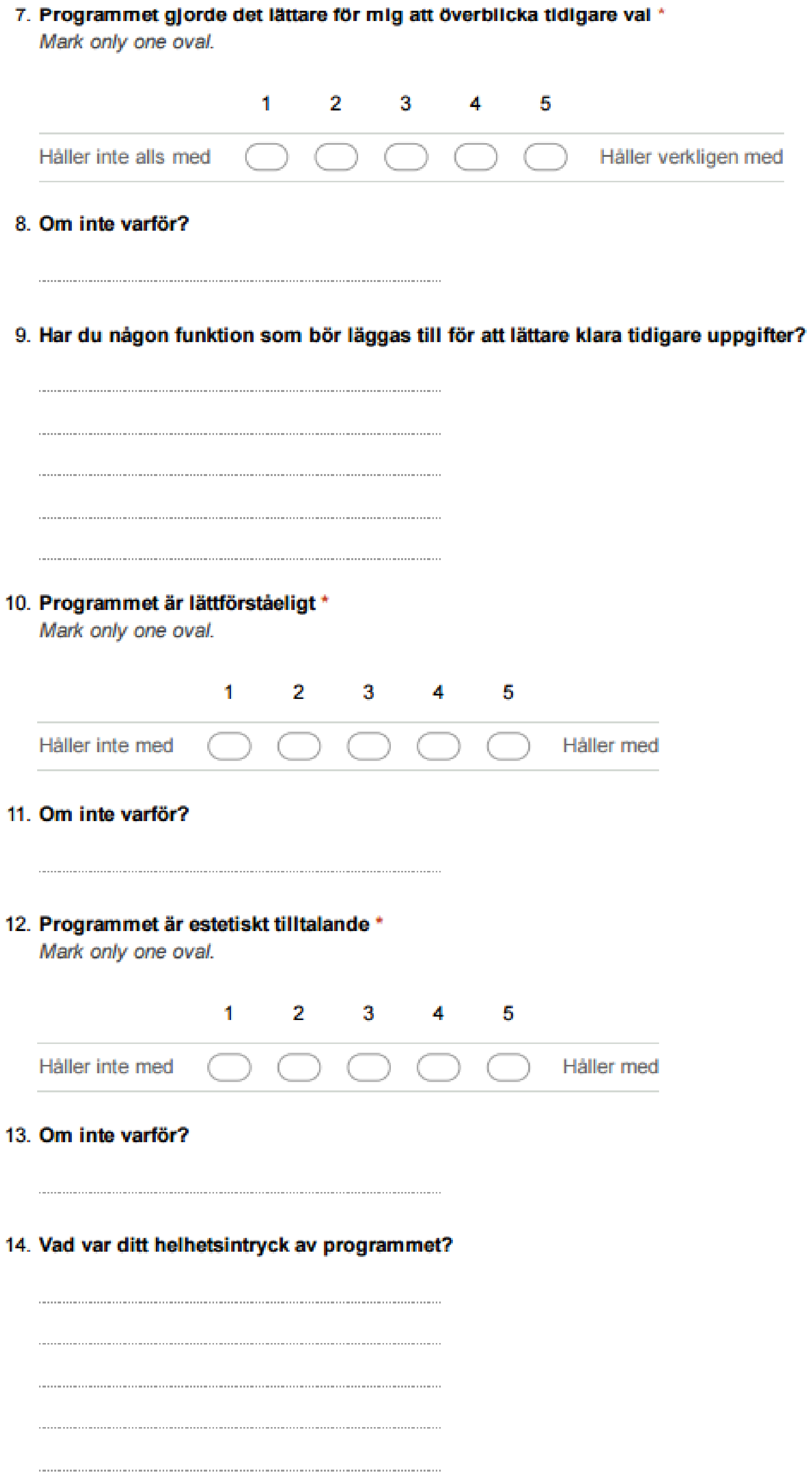
Mark only one oval.

	1	2	3	4	5	
Håller inte alls med	○	○	○	○	○	Håller verkligen med

5. **Uppgift 5 var enkel att utföra** *

Mark only one oval.

	1	2	3	4	5	
Håller inte alls med	○	○	○	○	○	Håller verkligen med



7. **Programmet gjorde det lättare för mig att överblicka tidigare val** *

Mark only one oval.

	1	2	3	4	5	
Håller inte alls med	○	○	○	○	○	Håller verkligen med

8. **Om inte varför?**

....

9. **Har du någon funktion som bör läggas till för att lättare klara tidigare uppgifter?**

....

....

....

....

....

10. **Programmet är lättförståeligt** *

Mark only one oval.

	1	2	3	4	5	
Håller inte med	○	○	○	○	○	Håller med

11. **Om inte varför?**

....

12. **Programmet är estetiskt tilltalande** *

Mark only one oval.

	1	2	3	4	5	
Håller inte med	○	○	○	○	○	Håller med

13. **Om inte varför?**

....

14. **Vad var ditt helhetsintryck av programmet?**

....

....

....

....

....